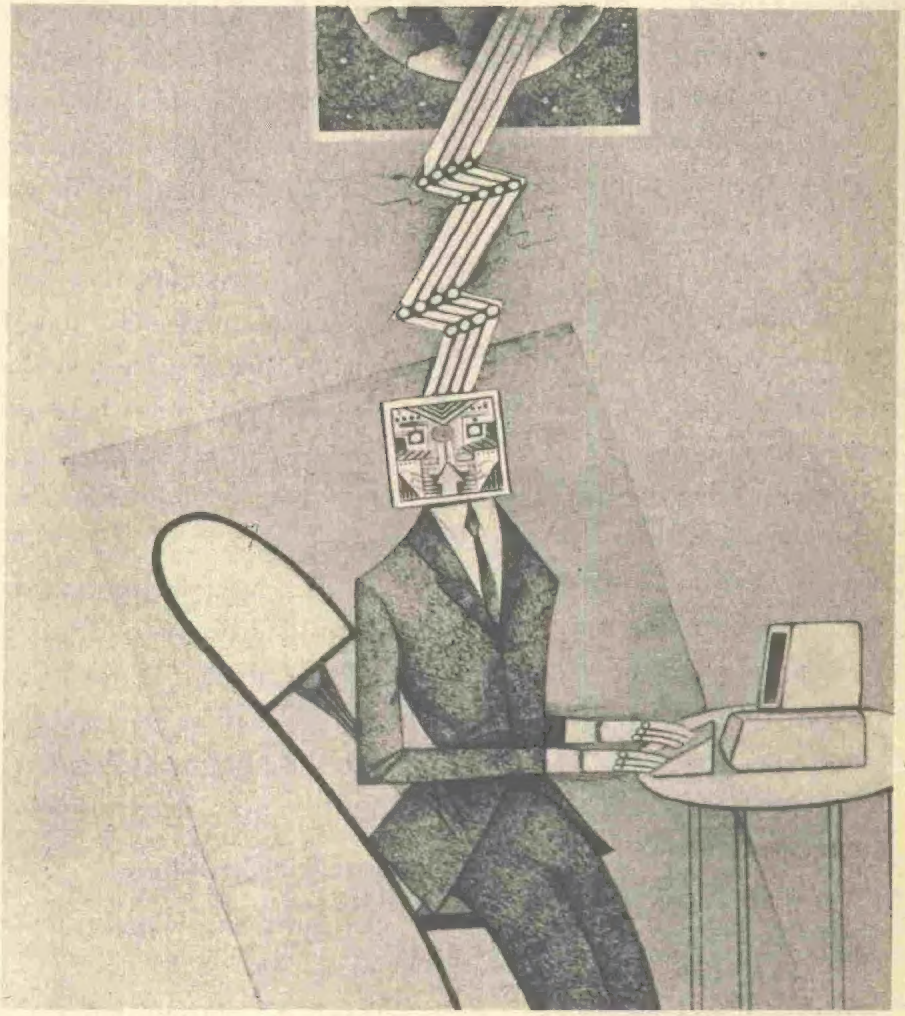
ซอฟต์แวร์

พันธุ์ปิติ เปี่ยมสง่า

SOFT MAIL

มาสร้าง สกรีนเอดิเตอร์ กันเถอะ

กล่าวถึง
การสร้างสกรีนเอดิเตอร์
เฉพาะทฤษฎีในการออกแบบ
พร้อมทั้งยกตัวอย่าง



ที่แน่นอนอย่างหนึ่งสำหรับระบบปฏิบัติการทุกแบบจะต้องมีเอดิเตอร์มาให้ด้วย เนื่องจากความจำเป็นของเอดิเตอร์ในการพัฒนาระบบมีความสำคัญมาก ตัวอย่างเช่น ถ้าเราต้องการใช้งานระบบจัดการฐานข้อมูล การป้อนข้อมูลหรือค้นหาข้อมูลจะต้องมีเอดิเตอร์เป็นส่วนประกอบ หรือว่าเวิร์ดโปรเซสเซอร์คงจะไม่เรียกว่าเวิร์ดโปรเซสเซอร์แน่ถ้าไม่มีเอดิเตอร์ การเขียนโปรแกรมเพื่อพัฒนาระบบงานต่าง ๆ ไม่ว่าใช้โปรแกรมภาษาใดก็ตามย่อมจะต้องมีเอดิเตอร์เป็นเครื่องมือที่สำคัญสำหรับการพัฒนา เอดิเตอร์จึงมีประกอบมาด้วยสำหรับคอมพายเลอร์ทุกตัว

ลักษณะของเอดิเตอร์ที่พบโดยทั่วไปมีอยู่ 2 ลักษณะคือ ไลน์เอดิเตอร์หรือเอดิเตอร์แบบแก้ไปทีละบรรทัด (line editor) และสกรีนเอดิเตอร์ (screen editor) ซึ่งเป็นเอดิเตอร์ที่แก้ไขได้เต็มจอภาพ สำหรับไลน์เอดิเตอร์ส่วนใหญ่มีมาให้พร้อมกับระบบปฏิบัติการ ในระบบปฏิบัติการยูนิกซ์จะมีไลน์เอดิเตอร์มาด้วย ซึ่งชื่อว่า ed และใน DOS มีชื่อว่า EDLIN.COM ไลน์เอดิเตอร์เป็นโปรแกรมซึ่งไม่ค่อยมีผู้นิยมใช้เนื่องจากว่าใช้ยากและไม่สามารถเห็นข้อมูลและแก้ไขในเวลาเดียวกัน เพราะต้องใช้คำสั่งควบคุมได้ทีละบรรทัดเท่านั้น ผู้ใช้ส่วนใหญ่จะใช้เมื่อไม่มีสกรีนเอดิเตอร์ สกรีนเอดิเตอร์ในระบบปฏิบัติการบางตัวเช่นยูนิกซ์จะให้มาด้วยโดยมีชื่อว่า vi และ emacs สำหรับดอส เอดิเตอร์ที่มีความสามารถสูงมีอยู่หลายตัว ส่วนหนึ่งคือเอดิเตอร์ที่มีมากับคอมพายเลอร์ ส่วนหนึ่งเป็นเอดิเตอร์ที่ผลิตขายเฉพาะ เช่น Brief และ Quick Edit ความสามารถของเอดิเตอร์เหล่านี้สูงมาก โดยทุกคีย์บนคีย์บอร์ดเป็นคีย์คำสั่งเกือบทั้งหมด และอีกส่วนหนึ่งคือโปรแกรมเวิร์ดโปรเซสเซอร์

ในบทความนี้จะกล่าวถึงการสร้างสกรีนเอดิเตอร์ เฉพาะทฤษฎีในการออกแบบ พร้อมทั้งยกตัวอย่างในบางส่วน ซึ่งประกอบด้วยความสามารถต่อไปนี้

1) ป้อนตัวอักษรเป็นข้อมูลรหัสแอสกีจากคีย์บอร์ด
2) ลบตัวอักษร
3) ลบบรรทัด
4) เลื่อนหน้าจอขึ้นลง, ซ้ายขวา
5) เปลี่ยนหน้าจอขึ้นลง
6) ค้นหาข้อความและเปลี่ยนข้อความ

สำหรับโปรแกรมตัวอย่างเป็นโปรแกรมภาษาซี ซึ่งคอมพายล์ด้วยเทอร์โบซีใช้งานกับดอส เหตุที่ใช้กับคอมพายเลอร์ตัวอื่นไม่ได้เนื่องจากการกำหนดชื่อฟังก์ชันในการใช้คำสั่งในไบออสและดอสที่ไม่เหมือนกัน แม้ในบางคำสั่งของไมโครซอฟต์ซีไม่มีตัวอย่างของฟังก์ชันที่ทำงานแบบเดียวกันแต่กำหนดชื่อไม่เหมือนกัน เช่น คำสั่งในการอ่านค่าคีย์บอร์ดโดยไบออสในเทอร์โบซีมีชื่อว่า bioskey ( ) ส่วนไมโครซอฟต์ซีชื่อว่า __bios__keybrd ( ) และฟังก์ชันที่ไมโครซอฟต์ไม่มีเช่น คำสั่ง clrscr ( ) ซึ่งเป็นคำสั่งในการลบจอภาพของเทอร์โบซี

ทฤษฎีที่สำคัญบางอย่างสำหรับสกรีนเอดิเตอร์

ถึงแม้ว่าสกรีนเอดิเตอร์จะไม่ใช้งานเขียนซอฟต์แวร์ที่ยากนัก แต่ก็มีสิ่งที่น่าสนใจและไม่ว่าเขียนด้วยอัลกอริทึมแบบใดก็ยังจำเป็นต้องพิจารณาในหัวข้อต่อไปนี้

1. ตำแหน่งปัจจุบัน

สิ่งที่สำคัญและเป็นเรื่องที่ต้องคำนึงถึงมากที่สุดคือความสัมพันธ์กันระหว่างจอภาพ, ข้อมูลในหน่วยความจำ และข้อมูลบนดิสก์ เนื่องจากโครงสร้างของข้อมูลในแต่ละแบบแตกต่างกันโดยลักษณะของข้อมูลบนจอภาพจะเป็นอะเรย์ของบรรทัดและแถว ข้อมูลบนดิสก์เป็นสกรีนของไฟล์ซึ่งเป็นข้อมูลแบบอนุกรม ส่วนข้อมูลบนหน่วยความจำก็จะเป็นตามแบบที่โปรแกรมเมอร์เลือกใช้

อีกประการหนึ่ง ข้อมูลบนจอภาพเป็นการแสดงข้อมูลเพียงส่วนหนึ่งของไฟล์เท่านั้น ขณะที่โปรแกรมจะต้องเก็บข้อมูลของไฟล์ทั้งหมด โดยข้อมูลบางส่วนของไฟล์อ่านข้อมูลมาไว้บนหน่วยความจำขณะที่เนื้อไฟล์ทั้งหมดยังอยู่ในดิสก์

ดังนั้นตำแหน่งปัจจุบันจึงเป็นสิ่งที่จะต้องคำนึงถึงเป็นอันดับแรก เพราะว่าเป็นสิ่งที่จะบอกได้ว่างานต่อไปของโปรแกรมจะต้องทำอะไร และจะต้องสร้างความสัมพันธ์กันอย่างไรระหว่างข้อมูลทั้ง 3 แห่ง

2. การแทรกและลบตัวอักษร

การแทรกและลบตัวอักษรเป็นอีกสิ่งหนึ่งที่จะต้องคำนึงถึง แต่คราวนี้จะเป็นในแง่ประสิทธิภาพและความเร็วในการทำงาน การแทรกตัวอักษรในแง่ของหน่วยความจำคือ การย้ายข้อมูลไป 1 ตัวอักษรจากตำแหน่งปัจจุบันแล้วนำตัวอักษรที่ต้องการแทรกไปไว้แทน ส่วนการลบตัวอักษรคือการย้ายข้อมูลในตำแหน่งถัดไปจนกระทั่งถึงท้ายข้อมูลมาข้างหน้า ถ้าหากว่าข้อมูลมีขนาดใหญ่มากการแทรกหรือการลบตัวอักษรจะทำให้โปรแกรมช้าลงมาก

3. ข้อกำหนดของการทำงาน

เนื่องจากการแก้ไขข้อมูลจะต้องทำในบัฟเฟอร์ของข้อมูลตามที่โปรแกรมกำหนดไว้ การพิมพ์ข้อมูลเข้าไปจนกระทั่งเกินขนาดของบัฟเฟอร์โปรแกรมย่อมทำงานไม่ได้ สิ่งสำคัญของโปรแกรมเมอร์จะต้องพิจารณาว่าข้อกำหนดควรอยู่ที่ไหนหรือถ้าจะให้เกินกว่านี้ควรจะพิจารณาเพิ่มขนาดของบัฟเฟอร์อย่างไร

โครงสร้างข้อมูลของโปรแกรมตัวอย่าง

ในโปรแกรมตัวอย่างได้กำหนดบัฟเฟอร์เป็นบัฟเฟอร์ซึ่งมีโครงสร้างแบบอะเรย์ตามขนาดของบรรทัดที่มากที่สุดคือ 256 บรรทัด โดยแต่ละบรรทัดจะมีตัวอักษรได้ไม่เกิน 127 ตัวอักษร และมี linestr เป็นอะเรย์ของพอยน์เตอร์เพื่อชี้ตำแหน่งข้อมูลของแต่ละบรรทัดว่าเก็บไว้ที่บัฟเฟอร์ตัวไหน การกำหนดค่าในโปรแกรมกำหนดดังรูปที่ 1

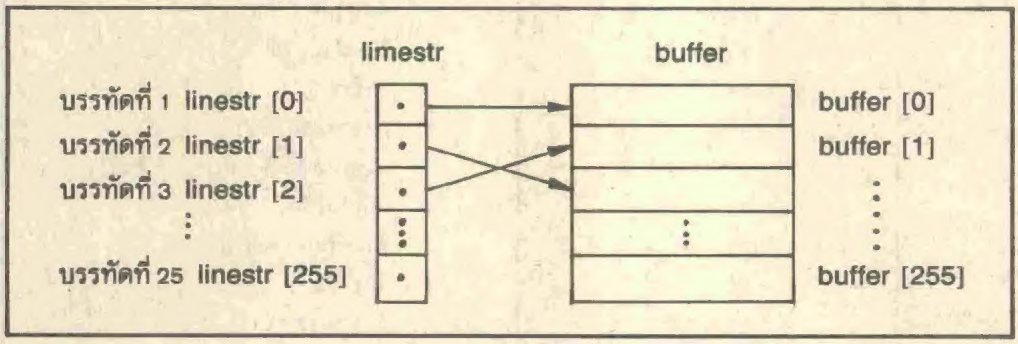
```
# define MAXCHAR 128
# define BLOCKLINE 256

unsigned char *linestr [BLOCKLINE];
unsigned char  buffer [BLOCK LINE] [MAXCHAR];
```

รูปที่ 1 การกำหนดโครงสร้างข้อมูล

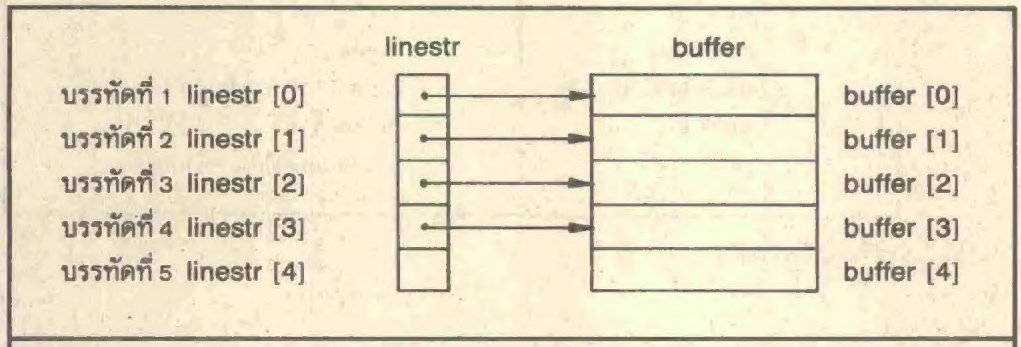
สำหรับในรูปที่ 2 เป็นการแสดงให้เห็นว่าเราสามารถปรับลำดับของบรรทัดให้ไม่ขึ้นกับลำดับของบัฟเฟอร์ได้โดยใช้ linestr ซึ่งเป็นอะเรย์ของพอยน์เตอร์มาชี้แทน ดังนั้นการแทรกบรรทัดและลบบรรทัดจึงไม่จำเป็นต้องย้ายข้อมูลจริง เพียงแต่เลื่อนพอยน์เตอร์ของการชี้บรรทัดเท่านั้น สมมติว่าข้อมูลเรามีอยู่ 4 บรรทัดและถ้ามีการแทรกข้อมูลในบรรทัดที่ 3 และลบข้อมูลในบรรทัดที่ 3 ตำแหน่งของ linestr จะเปลี่ยนไปดังรูปที่ 3

การอ่านค่าจากคีย์บอร์ด

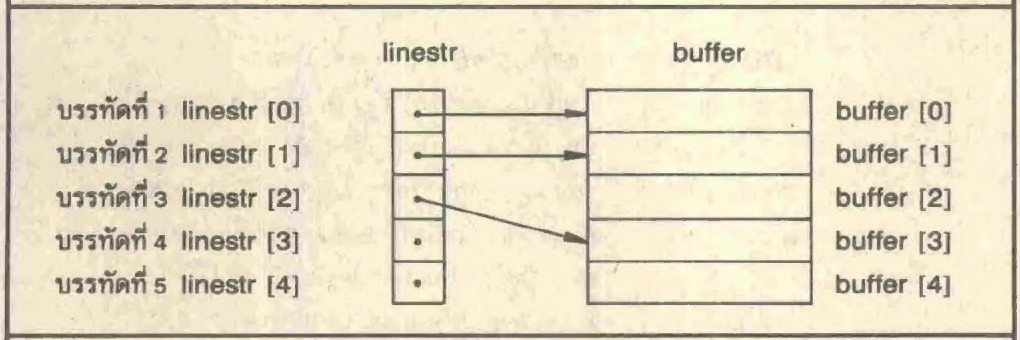
การอ่านค่าจากคีย์บอดร์ดในที่นี้ใช้การอ่านสแกนคีย์โดยตรง จากฟังก์ชัน bioskey ( ) ค่าที่ได้จากฟังก์ชันนี้ไบต์แรกจะเป็นค่าของคีย์ที่เป็นคำสั่ง ส่วน



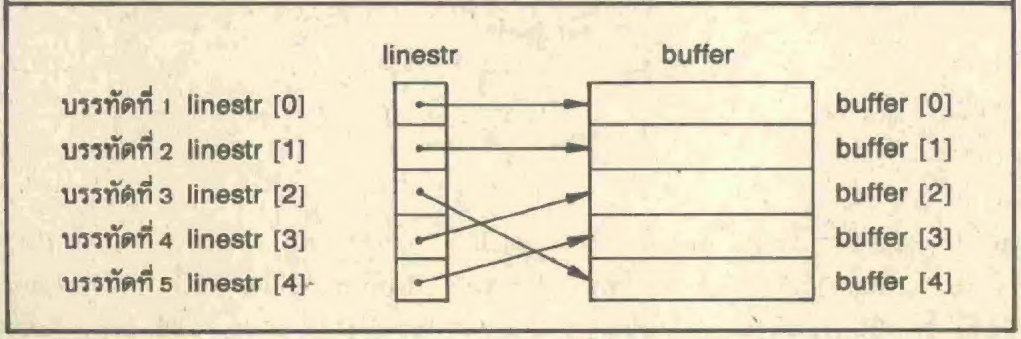
รูปที่ 2 ความสัมพันธ์ของแต่ละบรรทัดกับข้อมูลในบัฟเฟอร์



รูปที่ 3.1 ข้อมูลก่อนทำการแทรกและลบ



รูปที่ 3.2 ข้อมูลหลังการลบบรรทัดที่ 3



รูปที่ 3.3 ข้อมูลหลังการแทรกบรรทัดที่ 3

ไบต์หลังจะเป็นตัวอักษร การแปลงค่าสแกนคีย์เป็นตัวอักษรทำได้โดยนำค่าสแกนคีย์มาทำการ and กับ 0X00FF ก็จะได้ตัวอักษรออกมา

การทำงานของเอดิเตอร์ตัวอย่าง

เอดิเตอร์ตัวนี้ก็เหมือนกับเอดิเตอร์ที่พบกันทั่วไปคือจะทำงานด้วย main loop ซึ่งทำหน้าที่หลัก 2 ประการคือ อ่านค่าคีย์เข้ามาและนำค่าคีย์ที่อ่านได้มาตีความหมายว่าเป็นตัวอักษรหรือคำสั่ง สำหรับคำสั่งของเอดิเตอร์ตัวอย่างอยู่ที่รูปที่ 4

ก่อนที่จะอธิบายโปรแกรมขออธิบายตัวแปร global ที่สำคัญคือ xPOS, yPOS, line, col, maxline, topline และ topcol ซึ่งมีรายละเอียดดังรูปที่ 5

จะสังเกตเห็นได้อย่างหนึ่งคือ ในโปรแกรมจะต้องมีส่วนที่ควบคุมให้ตำแหน่งบนจอภาพตำแหน่ง xPOS, yPOS มีค่าตรงกับค่า linestr [line-1] [col-1] เสมอ (ที่ต้องลบออกจาก line และ col) เนื่องจาก line และ col เป็นหมายเลขบรรทัดและคอลัมน์เฉพาะแต่ในภาษาซี ตัวแรกของแต่ละอะเรย์คือหมายเลข 0 ดังนั้นจึงต้องลบด้วย 1 ทุกครั้งสำหรับการอ้างตำแหน่งหน่วยความจำ

สำหรับตำแหน่งลูปของโปรแกรมตัวอย่างนี้อยู่ในฟังก์ชัน ed ( ) โดยเริ่มต้นทำการเก็บสถานะค่าของจอภาพไว้ด้วยคำสั่ง gettextinfo ( ) เปิดไฟล์ด้วยคำสั่ง retrieve ( ) หลังจากนั้นเข้าลูปซึ่งจะทำการรับคีย์และตีความหมายคีย์ในกรณีที่คีย์ที่กดให้เป็นคีย์คำสั่ง ซึ่งเป็น default: ใน switch ของการเช็คคีย์จะทำการตรวจสอบว่าเป็นการทำงานแบบพิมพ์แทรก (insert) หรือพิมพ์ทับ (overwrite) ถ้าเป็นการพิมพ์แทรกจะจัดการเลื่อนตัวอักษรบนจอภาพไปทางขวาคอลัมน์และย้ายข้อมูล

คีย์	ความหมาย
ลูกศรซ้าย	เลื่อนไปทางซ้าย 1 ตัวอักษร
ลูกศรขวา	เลื่อนไปทางขวา 1 ตัวอักษร
ลูกศรลง	เลื่อนลงล่าง 1 บรรทัด
ลูกศรขึ้น	เลื่อนขึ้นบน 1 บรรทัด
HOME	ไปต้นบรรทัด
PgUp	กลับไปหน้าที่แล้ว
PgDn	ไปหน้าถัดไป
F1	Help
F2	แสดงจอภาพใหม่
F3	เปลี่ยนสีตัวอักษร
F4	เปลี่ยนสีพื้น
F5	อ่านข้อมูลจากไฟล์
F6	เก็บไฟล์
F7	ค้นหาคำ
F8	ค้นหาและเปลี่ยนค่า
ALT_D	ลบบรรทัด
ALT_X	ยกเลิกการทำงาน
BACKSPACE	ลบตัวอักษรด้านซ้าย 1 ตัวอักษร
RETURN	แทรกบรรทัด
DEL	ลบตัวอักษรที่ตำแหน่งเคอร์เซอร์

รูปที่ 4 คำสั่งของเอดิเตอร์

xPOS, yPOS	เป็นตัวอย่างปัจจุบันของเคอร์เซอร์
line, col	ตำแหน่งของตัวอักษรใน buffer ซึ่งปรากฏอยู่บนจอภาพ ตำแหน่ง XPOS, YPOS
maxline	จำนวน buffer ที่มากที่สุดที่ได้มีการ edit
topcol	ตำแหน่งของข้อมูลในแต่ละแถวที่แสดงบนตำแหน่งซ้ายสุดของจอภาพ
topline	หมายเลขบรรทัดที่แสดงบนบรรทัดแรกสุดของจอภาพ

รูปที่ 5

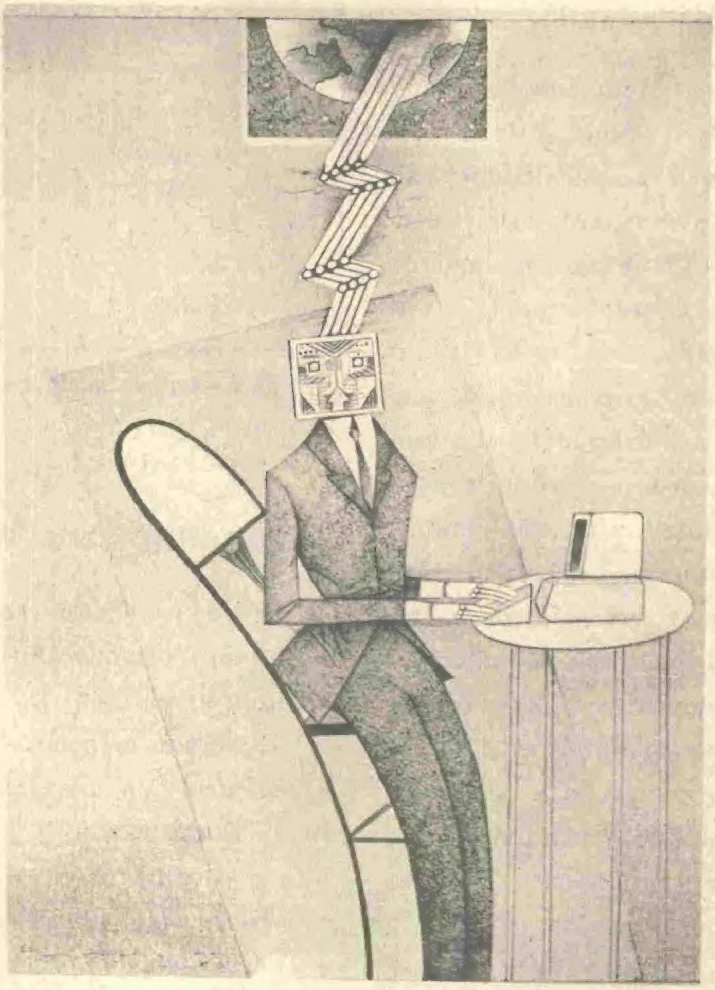
ในบัฟเฟอร์ไปทางขวา 1 ตำแหน่ง จากนั้นตรวจสอบดูว่าบรรทัดนั้นเป็นบรรทัดว่างหรือเปล่า ถ้าว่างให้เติมช่องว่างตั้งแต่คอลัมน์แรกจนถึงตำแหน่งปัจจุบัน จากนั้นวางตัวอักษรลงบนจอภาพและใส่ค่าในบัฟเฟอร์และถ้าตำแหน่งนั้นเป็นตำแหน่งขวาสุดของจอภาพให้เลื่อนตัวอักษรทั้งจอภาพไปทางซ้าย (Scroll left) 1 คอลัมน์และก็บนคอลัมน์ขวาสุดอีกครั้งหนึ่ง

การเลื่อนตำแหน่งของจอภาพและบัฟเฟอร์

จากที่กล่าวมาแล้วว่าสิ่งสำคัญของสกรีนเอดิเตอร์คือ ความสัมพันธ์ระหว่างจอภาพและหน่วยความจำ และข้อจำกัดตามข้อกำหนดของการทำงานที่ขนาดของจอภาพกับจำนวนตัวอักษรที่ไม่เท่ากัน ดังนั้นการเลื่อนตำแหน่งของจอภาพและตำแหน่งของบัฟเฟอร์จึงต้องแยกจากกัน โดยฟังก์ชันที่เกี่ยวกับจอภาพประกอบด้วย incX ( ), incY ( ), decX ( ), decY ( ) ทำหน้าที่เลื่อนตำแหน่งบนจอภาพและ incline ( ), inccol ( ), decline ( ), deccol ( ) ทำหน้าที่เลื่อนตำแหน่งในบัฟเฟอร์

การค้นหาข้อความ

การค้นหาข้อความเป็นการประมวลผลในหน่วยความจำเมื่อได้ผลลัพธ์แล้วจึงคำนวณตำแหน่งบนจอภาพ การทำงานของการค้นหาข้อความอยู่ในฟังก์ชัน __search (0) โดยต้องใส่พารามิเตอร์เป็น 0 เข้าไปด้วย การทำงานเริ่มโดยการอ่านข้อความที่ต้องการค้น จากนั้นทำการค้นหาโดยเปรียบเทียบหาตัวอักษรในแต่ละบรรทัดที่



เหมือนกับตัวอักษรตัวแรกของข้อความที่ต้องการหา เมื่อพบให้เปรียบเทียบข้อความที่ตำแหน่งนั้น ถ้าเท่ากันก็แสดงว่าพบแล้ว แต่ถ้าไม่เท่าให้เลื่อนจุดที่ต้องการเริ่มค้นไปเท่ากับข้อความที่ต้องการให้ค้นแล้วเปรียบเทียบตัวอักษรตัวแรกใหม่

การแทนข้อความ

หลังจากทำการค้นหาข้อความแล้ว ในกรณีที่เราต้องการแทนข้อความสามารถทำได้โดยย้ายข้อมูลที่ตำแหน่งท้ายข้อความที่พบไปในตำแหน่งที่เท่ากับความยาวของข้อความที่จะมาแทนที่และนำข้อความใหม่มาแทนที่ข้อมูลที่ว่างอยู่

การแทนข้อความเป็นการทำงานในฟังก์ชัน __search ( ) เช่นเดียวกับการค้นข้อความแต่ต้องใส่พารามิเตอร์เป็น 1

การอ่านไฟล์และเขียนไฟล์

การอ่านไฟล์อยู่ในคำสั่ง retrive ( ) ส่วนการเขียนไฟล์อยู่ในคำสั่ง save ( ) ทั้งการอ่านและเขียนไฟล์ทำในลักษณะเดียวกันคือ อ่านและเขียนทีละบรรทัดและจะจบด้วยรหัสท้ายไฟล์ (character 0x1A)

เวิร์ดโปรเซสเซอร์

สำหรับรายละเอียดของเอดิเตอร์ตัวอย่างในข้างต้นก็คงเพียงพอสำหรับการพิมพ์ข้อความ แต่ก็ยังไม่เป็นเอดิเตอร์ที่ดีนัก ยังห่างไกลกับคำว่าเวิร์ดโปรเซสเซอร์ สิ่งที่จำเป็นสำหรับเอดิเตอร์ในเวิร์ดโปรเซสเซอร์ นั่นคือ การจัดการข้อมูลเป็นบล็อก การจัดรูปแบบของบรรทัด สำหรับการก๊อบปี้ และการย้ายบล็อก ถ้าโครงสร้างข้อมูลแบบนี้ก็คงไม่ใช่เรื่องยาก การก๊อบปี้สามารถทำได้โดยเพิ่มจำนวนบรรทัดเข้าไปและคัดลอกข้อมูลจากต้นทางมา การย้ายบล็อกและการลบทำโดยการเลื่อนพอยน์เตอร์ของ linestr เพียงอย่างเดียว

ส่วนสุดท้ายของเอดิเตอร์ที่ยังขาดคือ ขนาดของไฟล์ที่ต้องการ edit มีขนาดขึ้นกับจำนวนบรรทัด ดังนั้นถ้าหากว่าไฟล์ที่ต้องการอ่านมีขนาดเล็กแต่ข้อมูลมีจำนวนบรรทัดมาก ไฟล์นั้นจะถูกตัดไปบางส่วน วิธีการที่นิยมใช้กันในการแก้ไขปัญหานี้คือการ Swap memory ซึ่งเป็นการนำหน่วยความจำส่วนที่ไม่ได้ใช้ลงหน่วยความจำสำรองไปเพื่อให้หน่วยความจำหลักมีที่ว่างพอที่จะนำข้อมูลส่วนอื่นที่ต้องการใช้ขึ้นมาด้วยการแสดงผล

เทอร์โบซี กับ ไมโครซอฟต์ซี

สำหรับผู้ที่ไม่ชอบใช้เทอร์โบซีและต้องการใช้ไมโครซอฟต์ซีแทนจะต้องเขียนฟังก์ชันบางฟังก์ชันขึ้นเองเนื่องจากไมโครซอฟท์ซีไม่มีไลบรารี่ไว้ให้ ฟังก์ชันดังกล่าวมีรายละเอียดในรูปที่ 6

ฟังก์ชัน	ความหมาย
clrsen ( )	ทำหน้าที่ลบจอภาพ
gotoxy (x,y)	ทำหน้าที่เลื่อน cursor ไปตำแหน่ง (x,y)
clreol ( )	ทำหน้าที่ลบจอภาพตั้งแต่ cursor ไปยังขวามือสุด
deccol ( )	ทำหน้าที่ลบบรรทัด 1 บรรทัด
inncol ( )	ทำหน้าที่เพิ่มบรรทัด 1 บรรทัด
movetex ( )	ทำหน้าที่ย้ายข้อมูลบนจอภาพ

รูปที่ 6 ฟังก์ชันที่เทอร์โบซีมี แต่ในไมโครซอฟต์ซีไม่มี

สกรีนเอดิเตอร์ภาษาไทย

การสร้างสกรีนเอดิเตอร์ภาษาไทยไม่ใช่โปรแกรมที่เขียนกันได้ง่าย ๆ เหตุผลหลายประการคือ

1. ต้องแสดงผลด้วยกราฟิกโหมดหรือการ์ดภาษาไทย หรือระบบภาษาไทยอื่น ๆ เช่น TKS

2. การเพิ่มตัวอักษร 1 ตัวสำหรับภาษาไทยไม่ได้หมายความว่าตำแหน่งของคอลัมน์จะเพิ่มขึ้น เนื่องจากตัวอักษรภาษาไทยมีหลายระดับ

3. ที่ตำแหน่งใด ๆ ในบรรทัดไม่สามารถบอกว่าอยู่ที่คอลัมน์ใดได้โดยการเปรียบเทียบกับต้นบรรทัดต้องมีอัลกอริทึมของภาษาไทยโดยเฉพาะ

4. การควบคุมตำแหน่งของเคอร์เซอร์กับตำแหน่งของบัฟเฟอร์ทำได้ยากในกรณีของการเลื่อนเคอร์เซอร์ด้วยลูกศรและ PgUp, PgDn

5. ต้องเขียนโปรแกรมโดยพิจารณาในแง่ความเร็วเป็นสำคัญเพราะการแสดงผลค่อนข้างช้า เนื่องจากอัลกอริทึมค่อนข้างยุ่งยากและโหมดการทำงานที่ไม่ใช่เท็กซ์โหมด

6. อื่น ๆ (ต้องลองเขียนเองแล้วจะรู้)

ก่อนจบ

โปรแกรมเอดิเตอร์ตัวอย่างนี้ได้เขียนขึ้นมาเพื่อเป็นแนวทางการศึกษาโครงสร้างและการทำงานที่เหมาะสมของสกรีนเอดิเตอร์ และเพื่อชี้จุดดีและจุดเสียต่าง ๆ ในแง่ของโครงสร้างข้อมูล การที่จะนำโปรแกรมนี้ไปใช้งานได้จึงต้องมีการพัฒนาต่อไป โดยซอฟต์แวร์ตัวนี้จึงมีอยู่ในซอฟต์เมล์ด้วย โดยไฟล์ข้อมูลมีชื่อว่า ED.C ทำการคอมพายล์โดยเทอร์โบซี

เอกสารอ้างอิง

Borland International Inc, TURBO C Reference guide and User Guide Version 1.5, 1987

Schildt M., Born to code in C, Osborne Mcgraw-Hill, 1989

ยืน ภู่วรวรรณ, สุรศักดิ์ สงวนพงษ์, โปรแกรมคอมพิวเตอร์ ภาษาแอสเซมบลี 8086/8088, ซีเอ็ดยูเคชั่น, 2530